

electronic design

sponsored by
 **National**
Semiconductor
The Sight & Sound of Information
 **www**®

A BLACK-BOX APPROACH To Analog-To-Digital Converters

Don Tuite, Analog/Power Editor

electronic
design  **eBook**

NEXT PAGE



- 03 • **Conversion Basics**
- 04 • **Encoding**
- 05 • **Straight Binary And Offset Binary Encoding**
- 06 • **Two's Complement Encoding**
- 07 • **LSB Precision**
- 08 • **Sampling And Conversion Time**
- 09 • **Aliasing**
- 10 • **Anti-Aliasing Filters**
- 11 • **A Summary Of ADC Specifications**

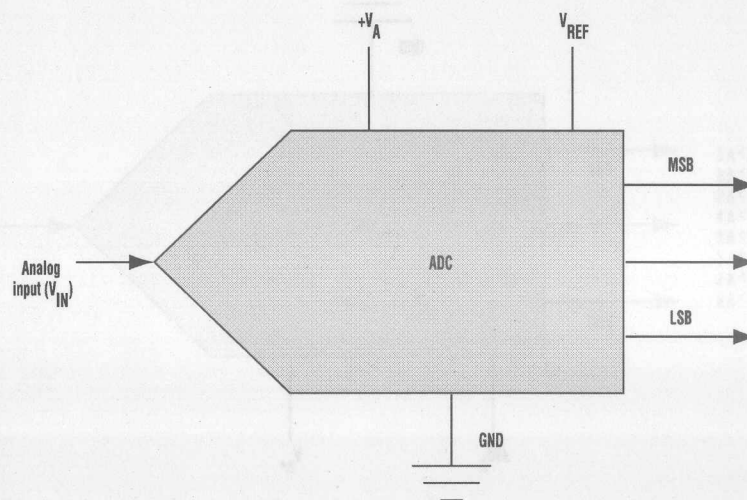


CONVERSION BASICS

Analog-to-digital converter (ADC) tutorials often focus on circuit details and their impact on performance. Such details include why a delta-sigma architecture provides precision at the expense of conversion rate, or why the flash architecture is fast but power-hungry.

In contrast, this eBook takes a black-box approach. It concentrates on the common characteristics of all ADCs and what they imply for the system-level designer.

This chapter provides some basic definitions and examines sample rate and resolution, bit weighting, encoding schemes, LSB value and resolution, reference voltage (V_{REF}), and output versus input. Subsequent chapters will deal with quantization error, noise, differential and integral nonlinearity (DNL and INL), missing codes, total unadjusted error (TUE), signal-to-noise ratio (SNR), signal-to-noise-plus-distortion (SINAD or SNDR), equivalent number of bits (ENOB), spur-free dynamic range (SFDR), total harmonic distortion (THD), and intermodulation distortion (IMD).



An ADC compares an analog signal at its input to a precision dc reference. It then generates a digital code on its output (in this case, a parallel bus, but it could be a serial interface) that represents the value of that analog signal within the precision allowed by that output word.

A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS:

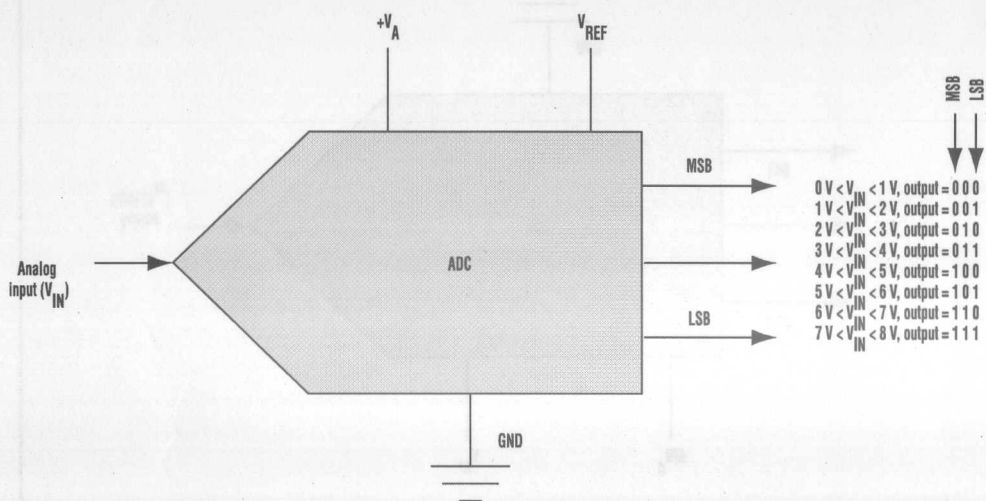
ENCODING

An ADC has an analog reference voltage or current against which the analog input is compared. The digital output word tells us what fraction of the reference voltage or current is the input voltage or current. So basically, the ADC is a divider.

The I/O transfer function is:

$$\text{Output} = 2^n \times G \times V_{\text{IN}}/V_{\text{REF}}$$

Where: n is the number of output bits (in other words, the resolution of the ADC), G is a gain factor (which is typically unity), V_{IN} is the analog input voltage, and V_{REF} or I_{REF} is the reference voltage or current. (In rare cases, the input and reference may be currents.)

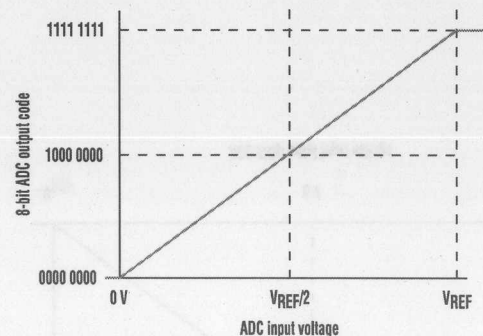


The resolution of the ADC above is 3 bits, which provides eight possible outputs. Assume V_{IN} is 5.5 V, V_{REF} is 8 V, and the gain is unity. The converter output would be 101, as it would be for any voltage between 5 and 6 V. The resolution does not permit any greater precision. (In practice, ADCs are made with a 1/2-LSB offset. More on this later.) In this case, the output is depicted as a parallel bus. Serial outputs are also used.

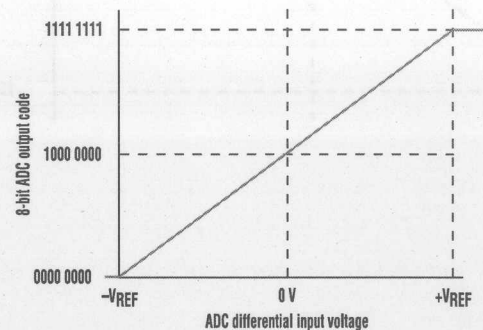


A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS: STRAIGHT BINARY AND OFFSET BINARY ENCODING

All ADCs have a top reference and a bottom reference. (The latter is usually, but not always, ground.) For an ADC with unity gain, the input range that can be digitized without significant error lies between these values, or between ground and V_{REF} . As an alternative, offset binary coding or two's complement encoding (next page) are sometimes used with differential-input ADCs.



With straight binary encoding, given a gain of $G = 1$, a V_{RN} (negative reference voltage) of 0 V, and an input voltage of 0 V, the output of an 8-bit ADC is 0000 0000. With an input of $V_{IN} = V_{REF}$, the 8-bit ADC output code is 1111 1111. With an input of $V_{IN} = V_{REF}/2$, the 8-bit ADC output code is 1000 0000.



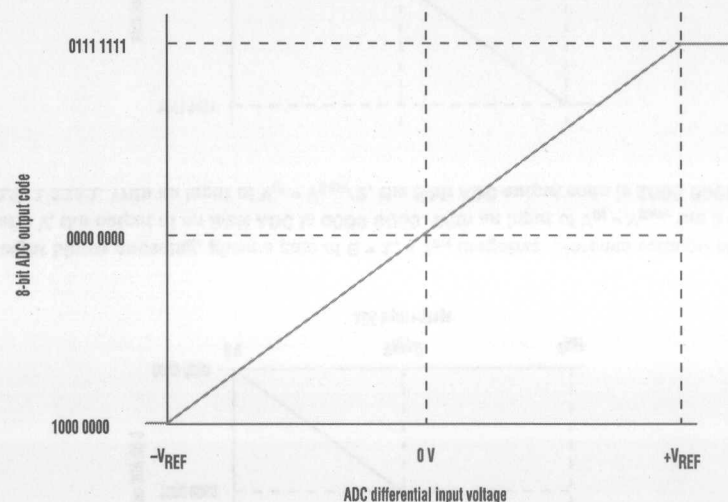
With offset binary encoding, the ADC output is at midscale when the differential input voltage is zero (+IN and -IN voltages are equal). It is all zeroes (negative full scale) when -IN is less than +IN by the value of V_{REF} , and it is all ones when +IN is greater than -IN by the value of V_{REF} . Thus, the full-scale input range is $2 \times V_{REF}$.



TWO'S COMPLEMENT ENCODING

Two's complement numbers are a way of encoding that accommodates both positive and negative numbers. With differential-input ADCs, it's particularly useful when calculations are to be performed on the ADC output word.

In two's complement notation, the MSB represents the sign of the quantity. That means that, with an 8-bit converter, the output could take on 127 values for which $-V_{IN}$ was less than $+V_{IN}$ and 128 values for which $-V_{IN}$ was greater than $+V_{IN}$.



In differential-input ADCs with two's complement encoding, the output for any given differential input is the same as for the ADCs with offset binary coding that were explained on the previous page. The only exception is the inverted value of the most significant bit.



LSB PRECISION

The value of one LSB is equal to $V_{REF}/2^n$, where n is the converter resolution.

Therefore, measurement precision can be increased both by employing a higher-resolution converter (more bits) and/or by shrinking the reference voltage.

Both of these approaches have drawbacks. Higher-resolution converters generally cost more and aren't available in speeds as high as lower-resolution converters.

One problem with reducing the reference voltage is a loss of input dynamic range. Another is greater sensitivity to input noise. Also, some industrial applications use relatively high-voltage transducers, which would need to be buffered to match the input requirements of the ADC.

LSB PRECISION AS DETERMINED BY REFERENCE VOLTAGE AND RESOLUTION

V_{REF}	Resolution	1 LSB
1.00 V	8	3.9062 mV
1.00 V	12	244.14 μ V
2.00 V	8	7.8125 mV
2.00 V	10	1.9531 mV
2.00 V	12	488.28 μ V
2.048 V	10	2.0000 mV
2.048 V	12	500.00 μ V
4.00 V	8	15.625 mV
4.00 V	10	3.9062 mV
4.00 V	12	976.56 μ V



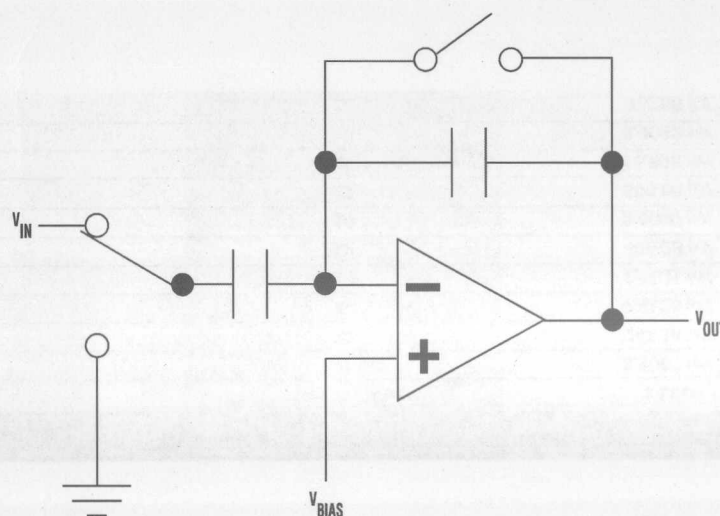
SAMPLING AND CONVERSION TIME

ADCs would be relatively uninteresting if they were nothing more than digital voltmeter components, making static measurements at random times.* Most ADCs are used to sample and digitize ac signals such as the down-converted IF in a radio receiver. Regular sampling requires a signal to clock the ADC at regular intervals.

An ADC's conversion time is the time it takes to convert the analog input signal to its equivalent digital code. It determines the maximum rate of change in the input signal that permits accurate conversion: $dV/dt_{MAX} = V_{REF}/(2^n \times \text{conversion time})$.

For example, if V_{REF} for a 10-bit ADC were 10.24 V, and conversion time were 10 ms, the maximum rate of change for conversion accuracy would be 1 V/s. If the input signal had a faster dV/dt , conversions would not be accurate.

To improve this amplitude uncertainty error, placing an analog amplifier that periodically latches the voltage on its input can reduce the effective aperture time. These are called sample-and-hold (or track-and-hold) circuits. They may be stand-alone chips, or they may be integrated within the ADC.



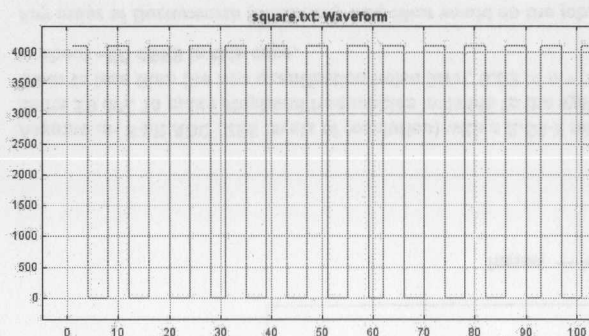
A simple S/H amplifier consists of a switch, hold capacitor, and an amplifier.

*This is an oversimplification for the purpose of introducing the topic of conversion time. Actually, there are some pretty interesting static ADC applications. Electronic weigh scales that make very precise weight measurements for process control in pharmaceutical manufacturing are one example. Such applications use slow but very high-resolution (up to 24 bits) ADCs.

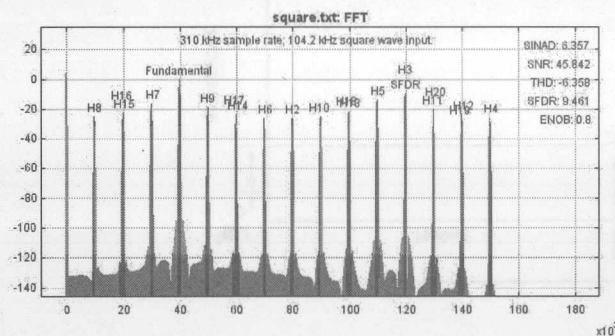


ALIASING

Shannon's and Nyquist's theorems tell us that any waveform can be accurately digitized by regularly sampling it at a rate of no less than twice the frequency of the waveform's highest frequency component. Of course, Fourier tells us that even a simple square wave has an infinite component series, but the real-world application generally defines where the limit can be drawn. (For example, in digitizing a radio signal, the baseband information to be recovered will have been bandwidth-limited.)



Any ADC is in part defined by its maximum sample rate. This in turn determines the highest allowable frequency component of the signal that can be applied without "aliasing." Aliasing means that a signal reconstructed from the data output of the ADC would contain frequency components not present in the original signal. The data would have been corrupted as higher-frequency components of the original signal were "folded back" to lower frequencies.

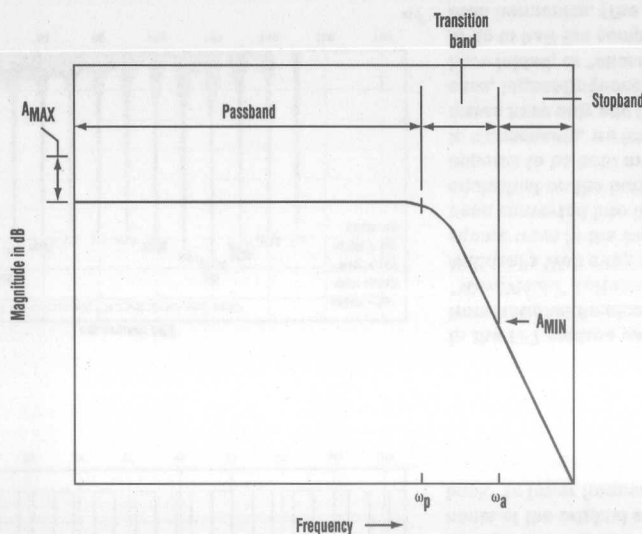


In the FFT capture on the left (obtained from National Semiconductor's "WaveVision" software, available from National's Web site), the time-domain square wave in the top waveform has been converted into its frequency domain equivalent on the bottom. Although there appears to be both even and odd harmonic components, we know that square waves have only odd harmonics. In this case, higher-frequency odd harmonics have folded, or "aliased," into the range of dc to half the sample rate to appear as even harmonics. (The WaveVision software provides dynamic performance parameters as indicated here. These parameters are meaningful only for sine wave inputs, so they have no meaning in the present context.)



ANTI-ALIASING FILTERS

Although it would be convenient simply to precede an ADC (or its sample-and-hold) with a low-pass filter with a cutoff frequency of one-half the ADC's sampling rate, (called the "Nyquist frequency"), it would be very difficult (and expensive) to build a filter with the necessary "brick-wall" frequency rolloff. Besides, such filters have severe group delay problems that can not be eliminated. Most designers are happy with filters that start to roll off around one-quarter of the sampling frequency.



Assume an 8-bit ADC (256 levels of resolution) with a 2.56-V reference so each quantization level, q , represents 10 mV. To make stopband frequencies invisible to the system, the anti-aliasing filter should attenuate them to less than the rms quantization noise level, $6.02 \times n + 1.76$ dB, or for 8 bits, $6.02 \times 8 + 1.76$ dB, which is about -50 dBFS in this case.

Any order of Butterworth (maximally flat) filter would do the job. But the higher the order, the faster the rolloff rate and the shorter the frequency gap between ω_p and ω_a on the filter magnitude response. (Where ω_p is the break point of the passband and ω_a is the point at which attenuation reaches -50 dBFS.) A complicating factor is the filter's group delay. Practical filter design requires tradeoffs of all these factors.



A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS:

A SUMMARY OF ADC SPECIFICATIONS

The chapters that follow will cover ADC specification in detail. To round out this chapter, here are brief definitions of key specs, distilled from National Semiconductor data sheets:

APERTURE DELAY is the time that elapses after the sampling edge of the clock to the point at which the input signal is acquired or held for conversion.

APERTURE JITTER is the variation in aperture delay from sample to sample. Aperture jitter shows up as noise in the conversion output.

BIT ERROR RATE (BER) is the probability of error and is defined as the probable number of errors per unit of time divided by the number of bits seen in that amount of time. A BER of 10^{-18} corresponds to a statistical error in one bit about every 31 years at 8 bits.

COMMON-MODE VOLTAGE is the common dc voltage applied to both input terminals of a differential input converter. This is always less than the voltage difference between the supply rails.

DIFFERENTIAL NON-LINEARITY (DNL) is the measure of the maximum deviation from the ideal step size of 1 LSB.

EFFECTIVE NUMBER OF BITS (ENOB, OR EFFECTIVE BITS) is a method of specifying signal-to-noise and distortion ratio, or SINAD. ENOB is defined as:

$$(\text{SINAD} - 1.76)/6.02$$

ENOB reduces a converter's precision below that promised by the absolute number of bits in its output word.

FULL-POWER BANDWIDTH (FPBW) is a measure of the frequency at which the reconstructed output fundamental drops 3 dB below its low-frequency value for a full-scale input.

GAIN ERROR is the deviation from the ideal slope of the transfer function. It can be calculated from offset and full-scale errors.

INTEGRAL NON-LINEARITY (INL) is a measure of the deviation of each individual code from a straight-line transfer function.

INTERMODULATION DISTORTION (IMD) is the creation of additional spectral components as a result of two sinusoidal frequencies being applied to the ADC input at the same time. It is defined as the ratio of the power in the second- and third-order intermodulation products to the power in one of the original frequencies. IMD is usually expressed in dBFS (dB referred to full-scale).

LATENCY is the number of clock cycles between initiation of conversion and when that data is presented to the output driver stage. New data is available at every clock cycle, but the data lags the conversion by the latency plus the output delay.



A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS:

A SUMMARY OF ADC SPECIFICATIONS

MISSING CODES are output codes that will never appear at the ADC outputs. These codes cannot be reached with any input value. Most of today's converters guarantee "no missing codes."

OUTPUT DELAY is the time delay after the rising or falling edge of the clock before the data update is presented at the output.

POWER-SUPPLY REJECTION RATIO (PSRR) measures how well the ADC rejects changes in the power-supply voltage. It can be the ratio of the change in full-scale error that results from a change in the dc power-supply voltage, expressed in dB. Or, it can be a measure of how well an ac signal riding upon the power supply is rejected at the output.

SIGNAL-TO-NOISE RATIO (SNR) is the ratio, expressed in dB, of the rms value of the input signal at the output to the rms value of the sum of all other spectral components below one-half the sampling frequency, not including harmonics or dc.

SIGNAL TO NOISE PLUS DISTORTION (S/(N+D) OR SINAD) is the ratio, expressed in dB, of the rms value of the input signal at the output to the rms value of all other spectral components below half the clock frequency, including harmonics but excluding dc.

SPURIOUS-FREE DYNAMIC RANGE (SFDR) is the difference, expressed in dB, between the rms values of the input signal at the output and the peak spurious signal, where a spurious signal is any signal present in the output spectrum that is not present at the input, excluding dc.

TOTAL HARMONIC DISTORTION is the ratio, expressed in dB, of the rms total of the first nine harmonic levels at the output to the level of the fundamental at the output.

SECOND HARMONIC DISTORTION is the ratio, expressed in dB, of the rms power in the input frequency seen at the output to the power in its second harmonic level at the output.

THIRD HARMONIC DISTORTION is the ratio, expressed in dB, of the rms power in the input frequency seen at the output to the power in its third harmonic level at the output.

electronic design

Sponsored by
 **National**
Semiconductor
The Spirit of Innovation

www.

A BLACK-BOX APPROACH To Analog-To-Digital Converters

Chapter 2

Don Tuite, *Electronic Design* and Nick Gray, *National Semiconductor*

electronic
design  **Book**

NEXT PAGE

+++ A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

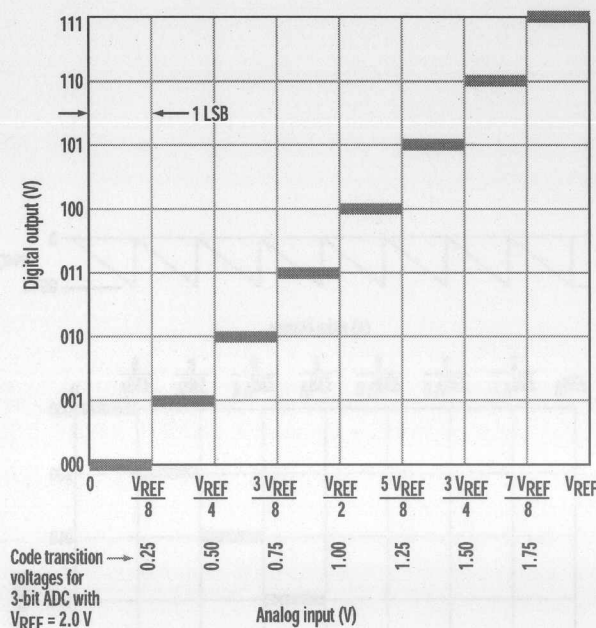
TABLE OF CONTENTS: CHAPTER 2: DC ERRORS

- 03 • Quantization**
- 04 • Quantization Error**
- 05 • Quantization Error With 1/2-LSB Offset**
- 06 • Differential And Integral Non-Linearity**
- 07 • Differential Non-Linearity And Missing Codes**
- 08 • Integral Non-Linearity**
- 09 • “End-Point” Vs. “Best-Fit” INL Measurements**
- 10 • Total Unadjusted Error (TUE)**
- 11 • Quantization Noise**
- 12 • End-Point Errors: Offset Error**
- 13 • End-Point Errors: Full-Scale Error**
- 14 • End-Point Errors: Gain Error**

QUANTIZATION

This chapter addresses dc error sources in analog-to-digital converters (ADCs). The critical terms include quantization, quantization error, linearity (DNL and INL), missing codes, end-point versus best-fit INL specifications, total unadjusted error (TUE), end-point errors, and quantization noise. All of these terms will be defined, and their implications for conversion accuracy will be discussed.

In the process of quantization, dc errors are inherent. Digitizing an analog quantity (voltage, current, temperature, pressure, etc.) involves assigning a range of analog values (a set of quanta, the latin plural of quantum) to a single digital code. That implies that all analog values that fall between some values x_m and x_n all will be assigned the same digital code at the output of the quantizer. All analog input values between another x_m and another x_n all will be assigned a different digital code. This gives rise to the stair-step transfer function of the ADC shown here.



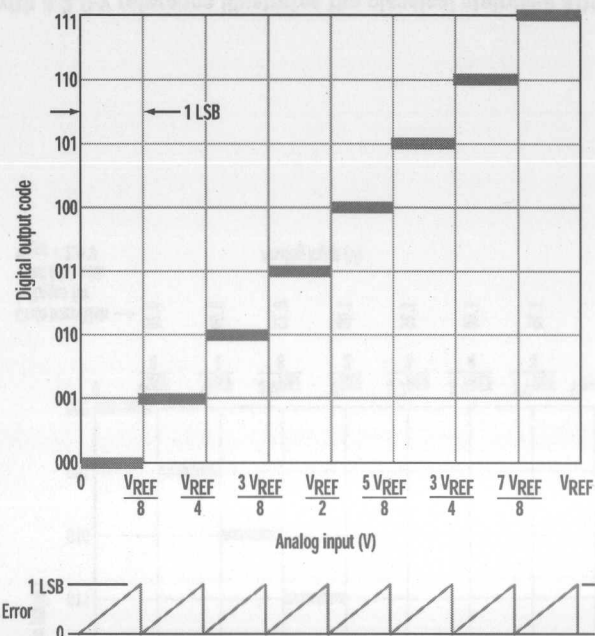
A simple 3-bit ADC with a 2.0-V reference illustrates the classical staircase ADC transfer function. The rising lines of the staircase are omitted to emphasize that the ADC output jumps from code to code. A gradual slewing from one digital code to another is not possible with a digital word. Because 3 bits provide a total of $2^3 = 8$ possible codes, each code transition occurs at intervals of $V_{REF}/8$. For a 2.0-V reference, these intervals, or quanta, are each $V_{REF}/8$, or $2.0/8 = 0.25\text{ V}$. (This hypothetical and somewhat coarse-grained 3-bit ADC with a 2-V reference will be used throughout this chapter, except where otherwise noted.)

QUANTIZATION ERROR

With an ADC input of zero, the output code of an ideal device should be zero (000), and there is no error. As the input voltage increases toward $V_{REF}/8$ ($V_{REF}/2^3$), the error increases because the input is no longer zero, but the output code remains at zero.

When the input reaches $V_{REF}/8$, the output code toggles from 000 to 001, where the output exactly represents the input voltage and the error reduces to zero.

As the input voltage increases past $V_{REF}/8$, the error again increases until the input voltage reaches $V_{REF}/4$, where the error again drops to zero. This process continues through the entire input range, and the error plot is a saw tooth.



Any given code could have been generated by a range of analog input values. Therefore, it is impossible to know what the precise input voltage was that produced that digital word. Another name for this is "quantization error." Quantization error is a round-off error. The maximum error, or "quantization uncertainty," in this example is 1 least significant bit (LSB).



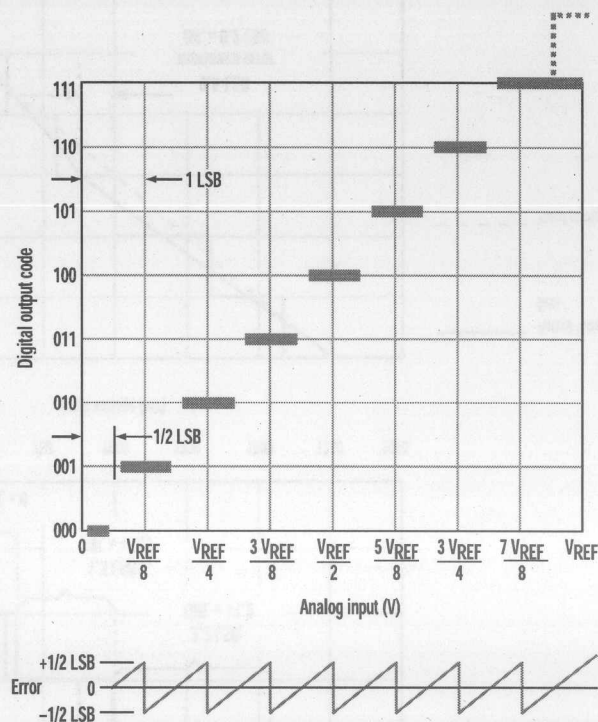
A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

QUANTIZATION ERROR WITH 1/2-LSB OFFSET

The 3-bit ADC with a $2\text{-}V_{REF}$ in these examples resolves the input into 0.25-V steps. So, it can be said either that the converter "resolution" is 8 bits, or that it is 0.25 V. That is, resolution can describe the number of bits the ADC has, and it can describe the size of the quantum corresponding to an LSB.

Note that an error of 0 to 1 LSB is not as desirable as an error of $\pm 1/2$ LSB. This is because an error range of 0 to 1 LSB means that the converter word has a maximum error of 1 LSB from the actual input value, whereas an error of $\pm 1/2$ LSB means that the converted word has a maximum error of just $1/2$ LSB from the actual input value.

The maximum output code corresponds to an input of 1 LSB less than the reference voltage. So a higher-resolution ADC will have a maximum output code that corresponds to an input level closer to the reference than would a lower-resolution ADC.



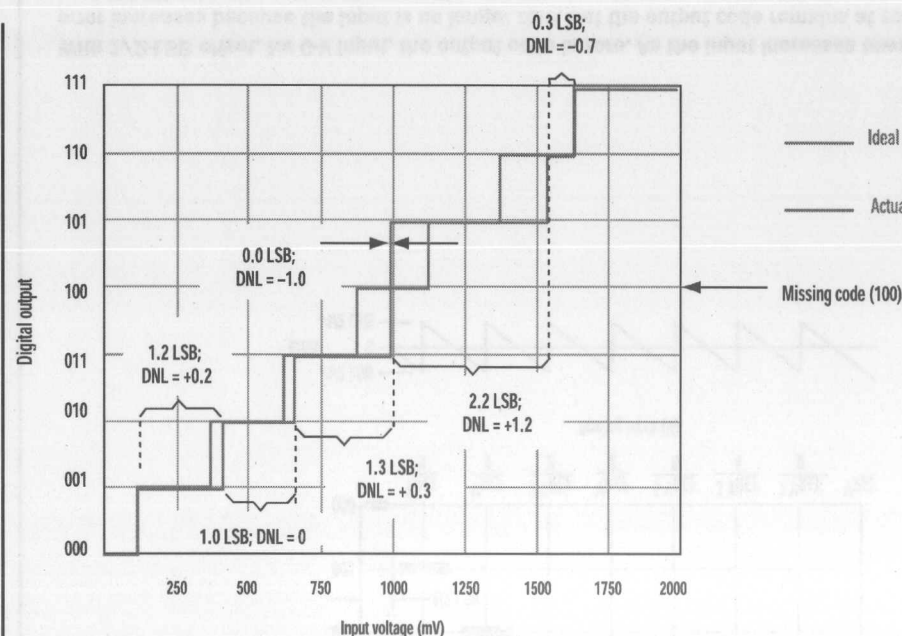
With 1/2-LSB offset, for 0-V input, the output code is zero. As the input increases toward the 1/2-LSB level, the error increases because the input is no longer zero, but the output code remains at zero. When the input reaches 1/2 LSB, the output code changes. Because the input is only at 1/2 LSB, the error is now $-1/2$ LSB. As the input increases, the error becomes less negative until the input reaches 1 LSB, where it is zero. As the input increases further, the error increases until it reaches $1-1/2$ LSB, where the output increments again and the sign of the error again becomes negative. This process continues through the entire input range.

A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

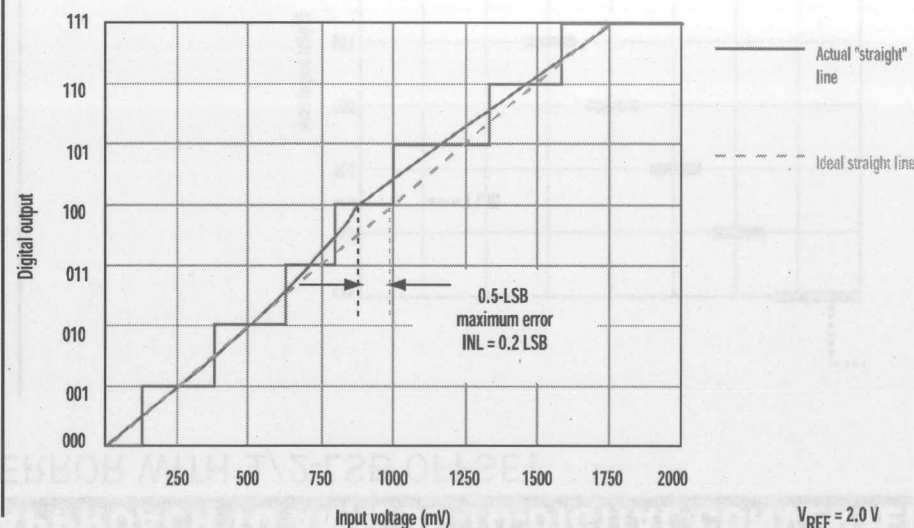
DIFFERENTIAL AND INTEGRAL NON-LINEARITY (DNL AND INL)

The terms DNL and DLE are both used to describe an error in step size. Similarly, INL and ILE are different terms used to describe the maximum deviation from the ideal transfer function.

The key to remembering the difference between these two specifications is the word "differential." DNL is the difference between the ideal and the actual code width, or quanta. The illustrations on the right will appear in a larger size on subsequent pages.



The widest input voltage range for a single code is the positive DNL specification. The narrowest input voltage range for a single code is the negative input DNL specification. These readings are converted to LSB, and then one LSB is subtracted from each to obtain the datasheet specification for positive and negative DNL.



INL is the maximum deviation of the transfer function from a straight line between two points along the input-output transfer curve. It is expressed in LSB.

$V_{REF} = 2.0 V$



A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

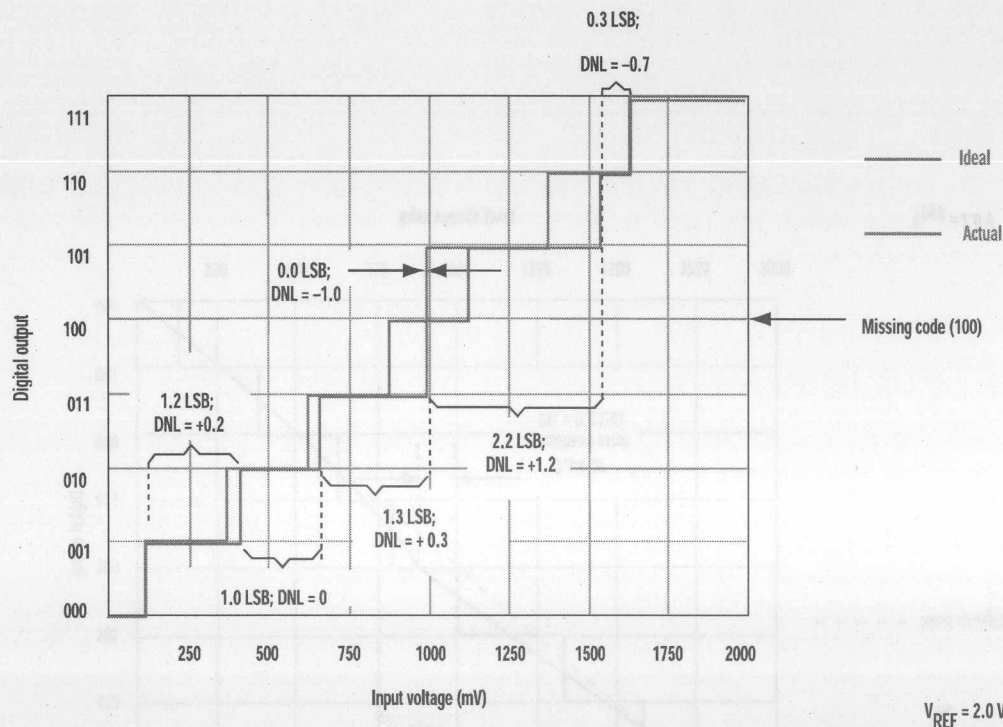
DIFFERENTIAL NON-LINEARITY AND MISSING CODES

In an ideal converter, the code-to-code transition points should be exactly 1 LSB apart and separated by $FSR/2^n$, where FSR is the full-scale input range and n is the converter resolution in bits.

In the illustration on the right, the transfer function in red represents a non-ideal 3-bit encoder. The caption describes how the datasheet specification for DNL is derived from the transfer function. It also introduces the concept of missing codes.

Many ADC data sheets specify “no missing codes.” Such a guarantee when the negative DNL spec is -1.0 means that the negative DNL cannot really reach -1.0 , but it can come so close that, for all practical purposes, the code in question does not really exist.

A negative DNL of -0.99 LSB means that the quanta for at least one code is just 1% of what it should be. This would mean that this quanta is so small that it virtually does not exist. For example, a 10-bit ADC with a gain of one and a 2.0-V reference would have a quanta for at least one code of just 19.5 μ V.



In this non-ideal quantization, a 250-mV input change causes the first transition (from 000b to 001b), so DNL for this step is zero. The second transition, from 001b to 010b, has an input change 1.2 LSB above the previous transition point, and the DNL for this step is $+0.2$ LSB. Similarly, the DNL for the third transition is zero. But look at what happens next. The output code does not change when the input changes from 1000 mV to beyond 1500 mV, so code 100b can never appear at the output. The output change that causes a transition from code 001b to code 100b is so small, it does not exist. So the DNL error is -1.0 LSB, and the code 100b is a missing code. To avoid missing codes in the transfer function, DNL must be more positive than -1.0 LSB. Continuing, the DNL for code 101b is $+1.2$ LSB, and code 110b has a -0.7 -LSB DNL. The DNL specification on the datasheet lists the worst positive and worst negative individual errors. In this case, the DNL specification would be -1.0 LSB, $+1.2$ LSB.

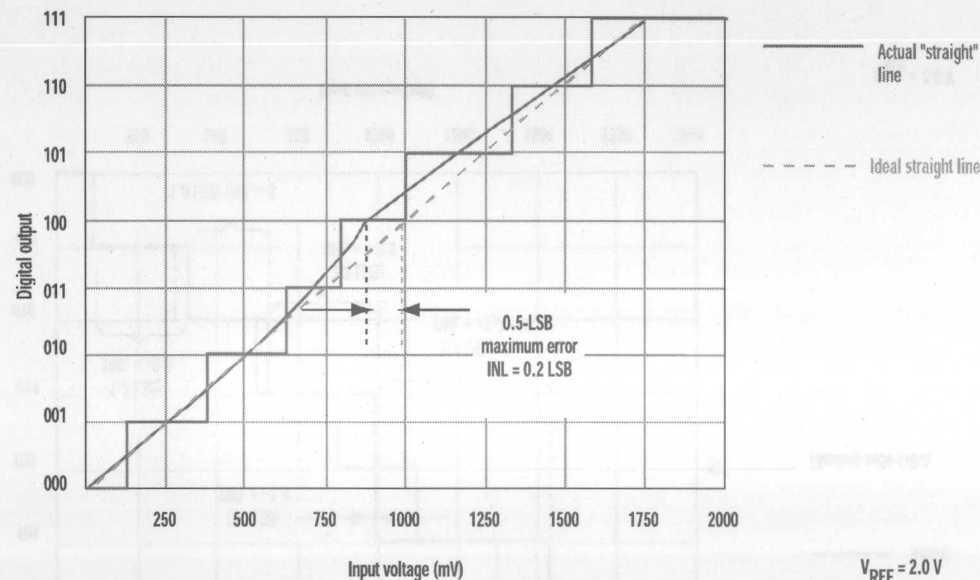
A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

INTEGRAL NON-LINEARITY

Interestingly, the size and distribution of the DNL errors determine the integral linearity of the converter, although the INL can be greater than the differential non-linearity.

INL is strictly a measure of the straightness of the transfer function. It does not include quantization errors, offset error, or gain error.

DNL and INL are static specifications. But they relate to two dynamic specifications, SNR in the case of DNL and THD in the case of INL. However, neither noise performance nor distortion can be predicted from the DNL or INL, but they tend to become worse as DNL or INL departs from zero.



INL is measured from the ideal centers of the code ranges to the actual centers of the code ranges. Sometimes a converter is described as "x bits linear." For instance, a converter with 10-bit resolution and 4-LSB non-linearity is sometimes described as an "8-bit linear" converter because 4 LSB for a 10-bit device is the same as 1 LSB for an 8-bit device. This arises out of the fact that, for a given reference voltage and gain factor, 1 LSB at 8 bits is the same code range as 4 LSB at 10 bits.

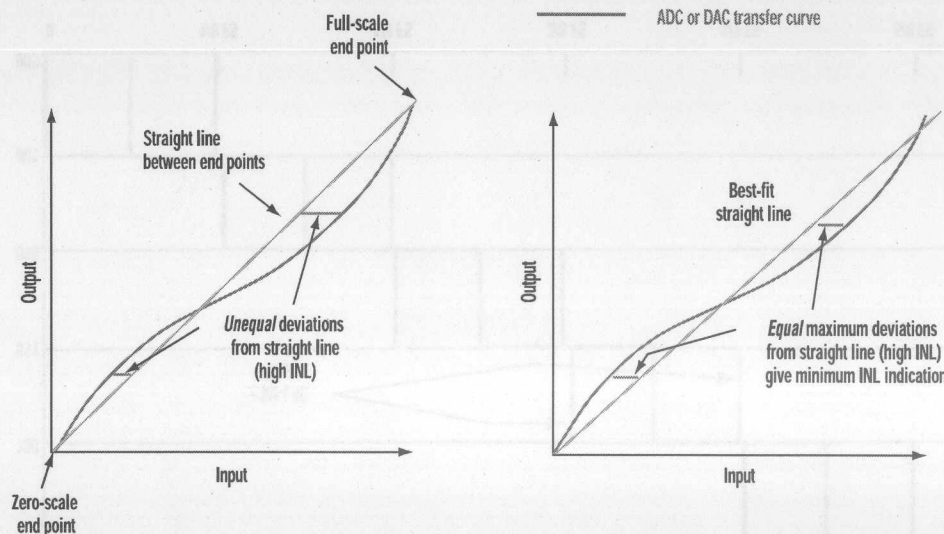
A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

"END-POINT" VS. "BEST-FIT" INL MEASUREMENTS

Designers can specify INL in one of two methods, shown in the diagrams to the right.

An end-point measurement is more conservative. It describes the worst-case INL that can be expected simply by considering the deviation of the actual transfer function from a straight line between the two end points. As the figure shows, a best-fit measurement paints a rosier picture.

Comparing the INL of two competing devices isn't reasonable unless the same measurement criterion is used for both.



One argument ADC vendors give for using the best-fit method in their data sheets is that board designers can adjust their circuits to actually realize that low INL, achieving better overall performance. The problem, though, is that each board must be adjusted for minimum INL for each individual converter—an expensive proposition for manufacturing.

Another argument says that best-fit is for dynamic applications only and that these applications aren't concerned with offset and gain errors (which are what cause the end-point and best-fit INL methods to diverge), unless the offset and gain errors are very large. This has some merit for dynamic applications, in that they are a better predictor of THD performance. But THD is usually specified anyway for converters intended for dynamic applications.

Generally, ADCs and DACs used in dc applications (which constitute most applications for general-purpose ADCs) should specify INL using the end-point method. But the method doesn't matter in strictly dynamic applications, as INL doesn't really affect them per se.

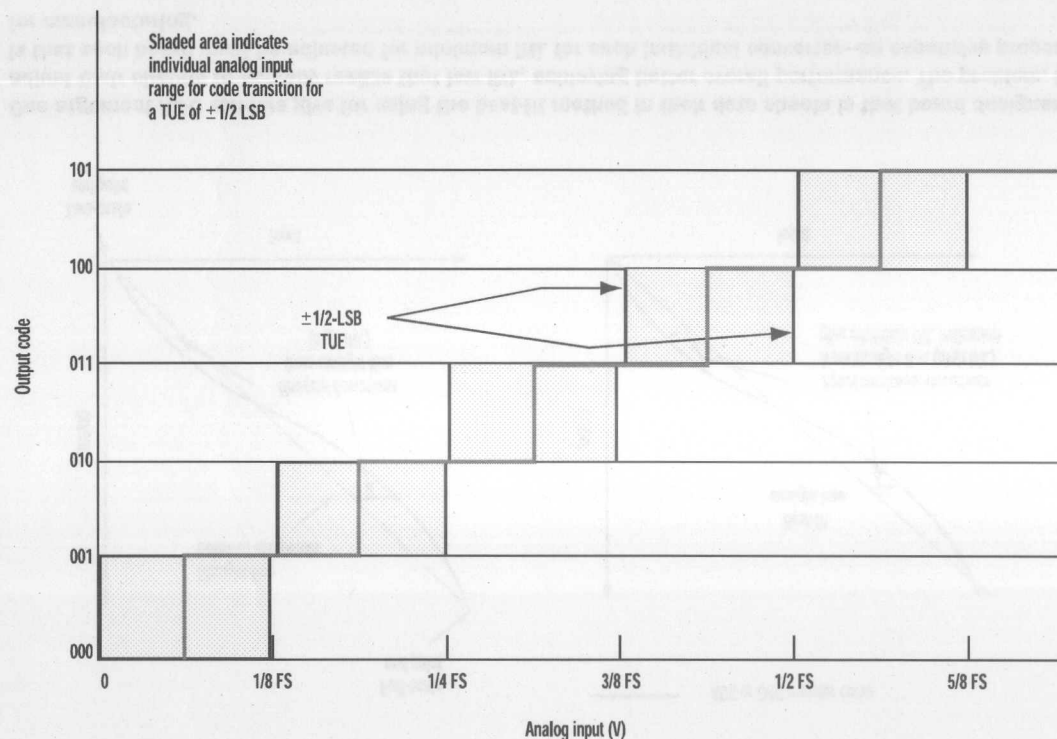
A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

TOTAL UNADJUSTED ERROR (TUE)

When a designer wishes to meet a specific error budget, total unadjusted error (TUE) provides a single specification that places a limit on errors from all sources. If this overall error limit is acceptable, no adjustments need to be made during manufacture of the end product.

Not all datasheets specify TUE. It's of value only when the total error specification is less than 1 or 2 LSB, so it generally isn't found on datasheets of converters with more than 8 bits of resolution.

If the total unadjusted error is much larger than about 1/2 LSB, it makes more sense to specify separate datasheet limits for offset error, gain error, and INL.

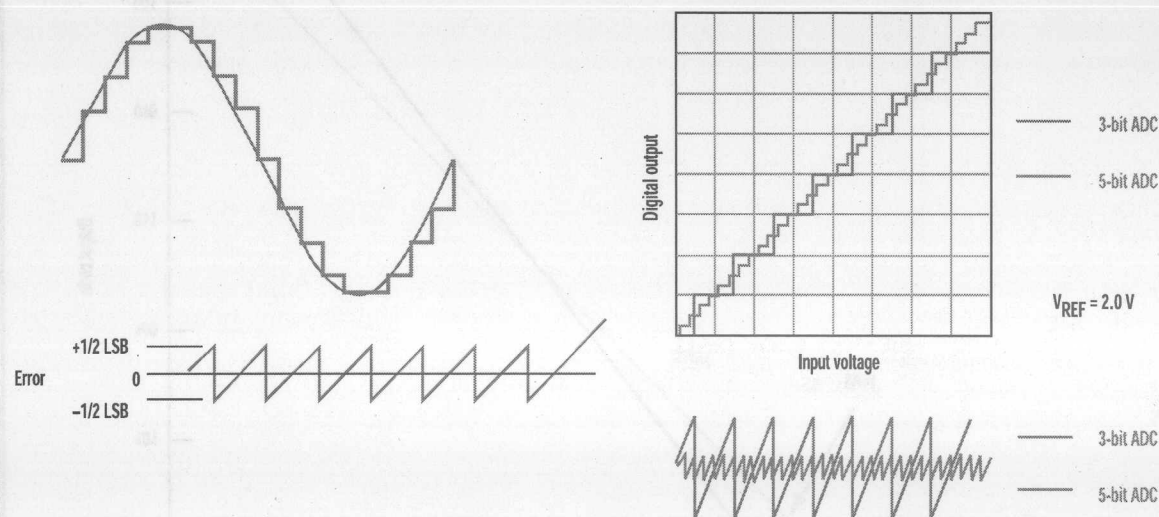


TUE consolidates linearity errors, gain error, and offset error. It's the worst-case deviation from the ideal device transfer function. TUE also is a static specification. It's useful for applications with dc or slowly moving input signals, such as digitizing the outputs of weigh scales and temperature and pressure sensors.

A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS:

QUANTIZATION NOISE

The act of quantizing the input signal adds noise to it. There is less quantization with higher-resolution converters because the input range is divided into a greater number of smaller ranges. This implies a higher maximum theoretical signal-to-noise ratio for higher-resolution converters.



The illustration on the left is a reminder that the error curve is a sawtooth wave in looking at the ADC error versus input code. The greater the area under this sawtooth wave, the more quantization noise is added to the signal. Adding more bits of resolution makes the step sizes (quanta) smaller, so the maximum error is smaller. Comparing the error plots on the right (for 3- and 5-bit ADCs) illustrates this point. Because the amount of quantization noise depends upon the area under the error curve, higher-resolution converters may offer better noise performance, other things being equal.

A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

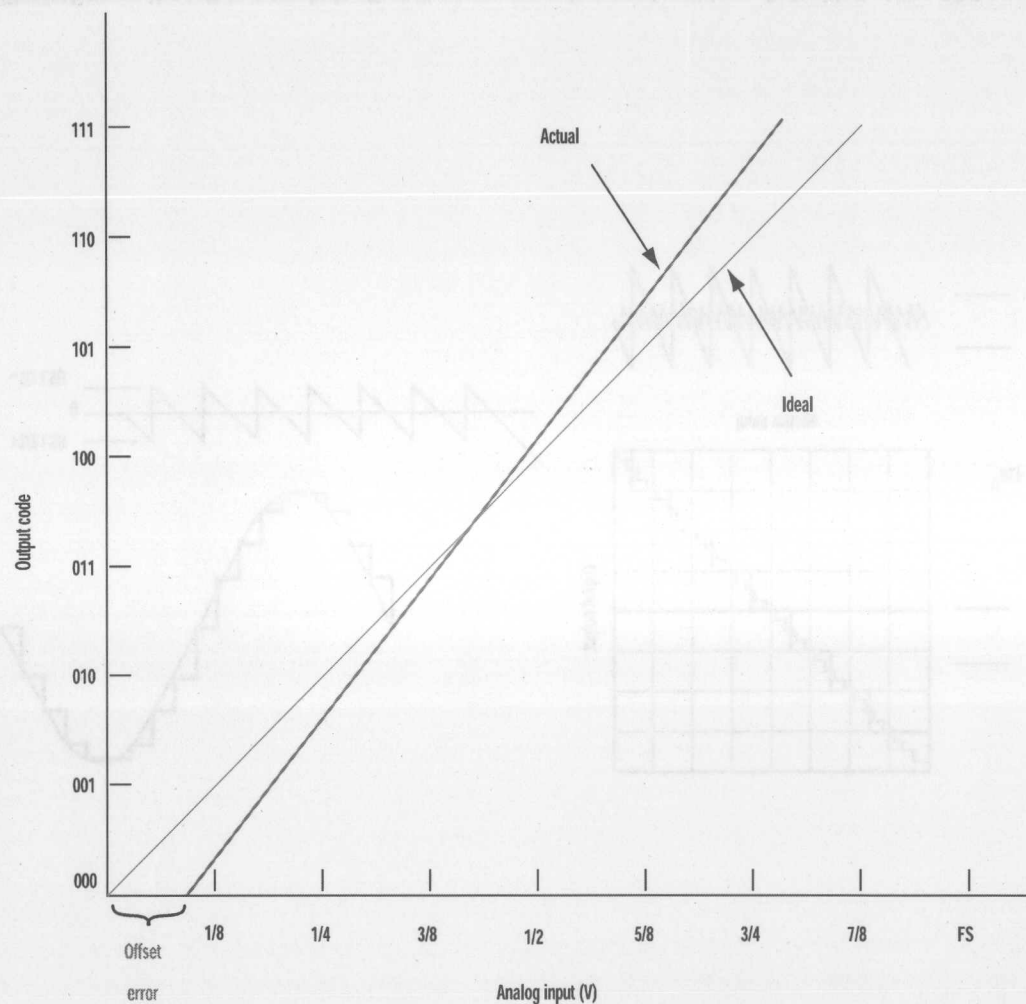
END-POINT ERRORS: OFFSET ERROR

There are three kinds of end-point errors, variously called offset error, full-scale error, and gain error or full-scale gain error.

In an ideal, single-ended input ADC, an input voltage of $q/2$ will just barely cause an output code transition from zero to a count of one. Any deviation from this is called zero-scale error, zero-scale offset error, or offset error.

Offset error for a differential input ADC is the error from mid-scale when both inputs are at the same potential.

Offset error may be expressed in volts, percent of full-scale voltage, or LSBs.



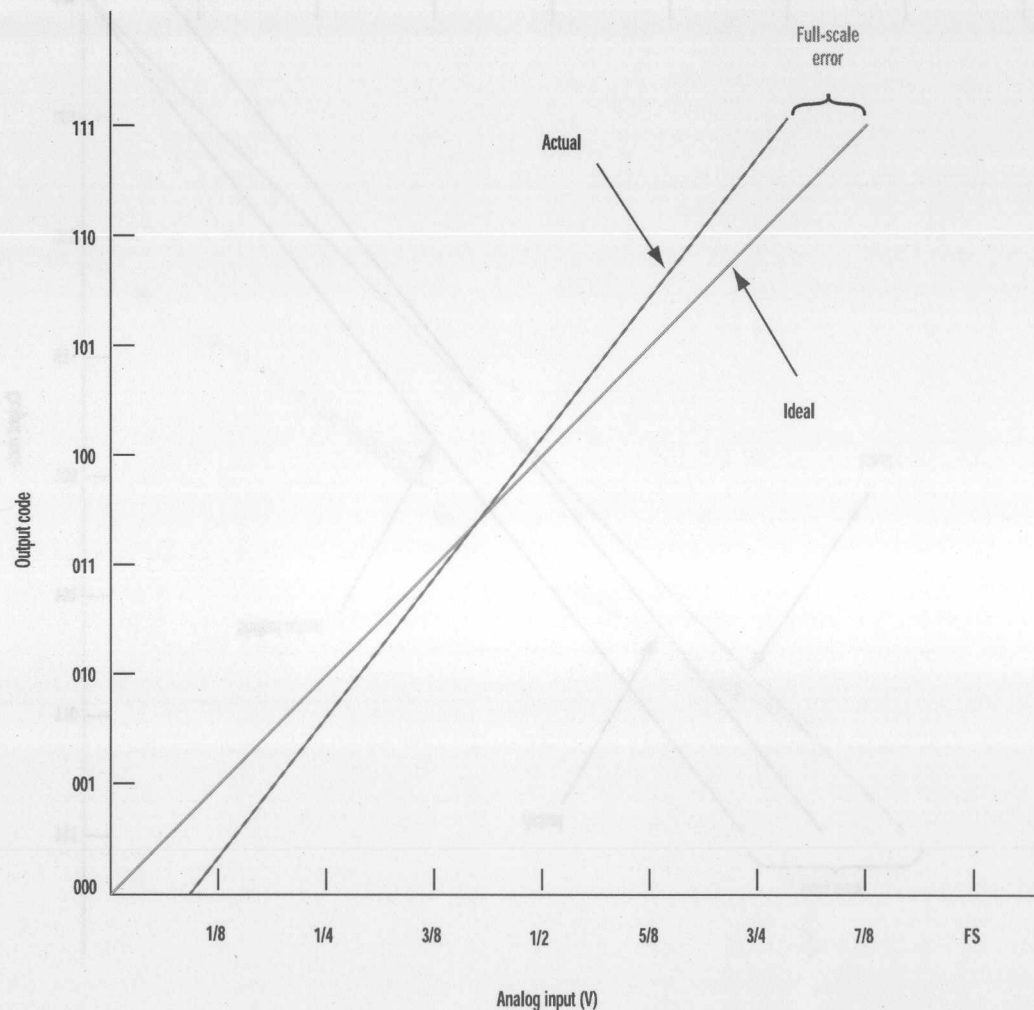
Offset error is positive when the input voltage required to cause the first output transition is higher than ideal and is negative when the input voltage required to cause this first transition is lower than ideal. Offset error is a constant, and it easily can be factored or calibrated out.



A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

END-POINT ERRORS: FULL-SCALE ERROR (FULL-SCALE OFFSET ERROR)

Full-scale error is the error in the full-scale output transition point. Part of this error will be due to offset voltage, and the rest will be due to an error in the slope of the transfer function. Full-scale error may be expressed in volts, as a percentage of the full-scale voltage, or in LSBs. Also, full-scale error is sometimes called full-scale offset error. Differential input ADCs have both positive and negative full-scale errors.



In an ideal, single-ended ADC, the output code transition to full scale just barely occurs when the input voltage equals

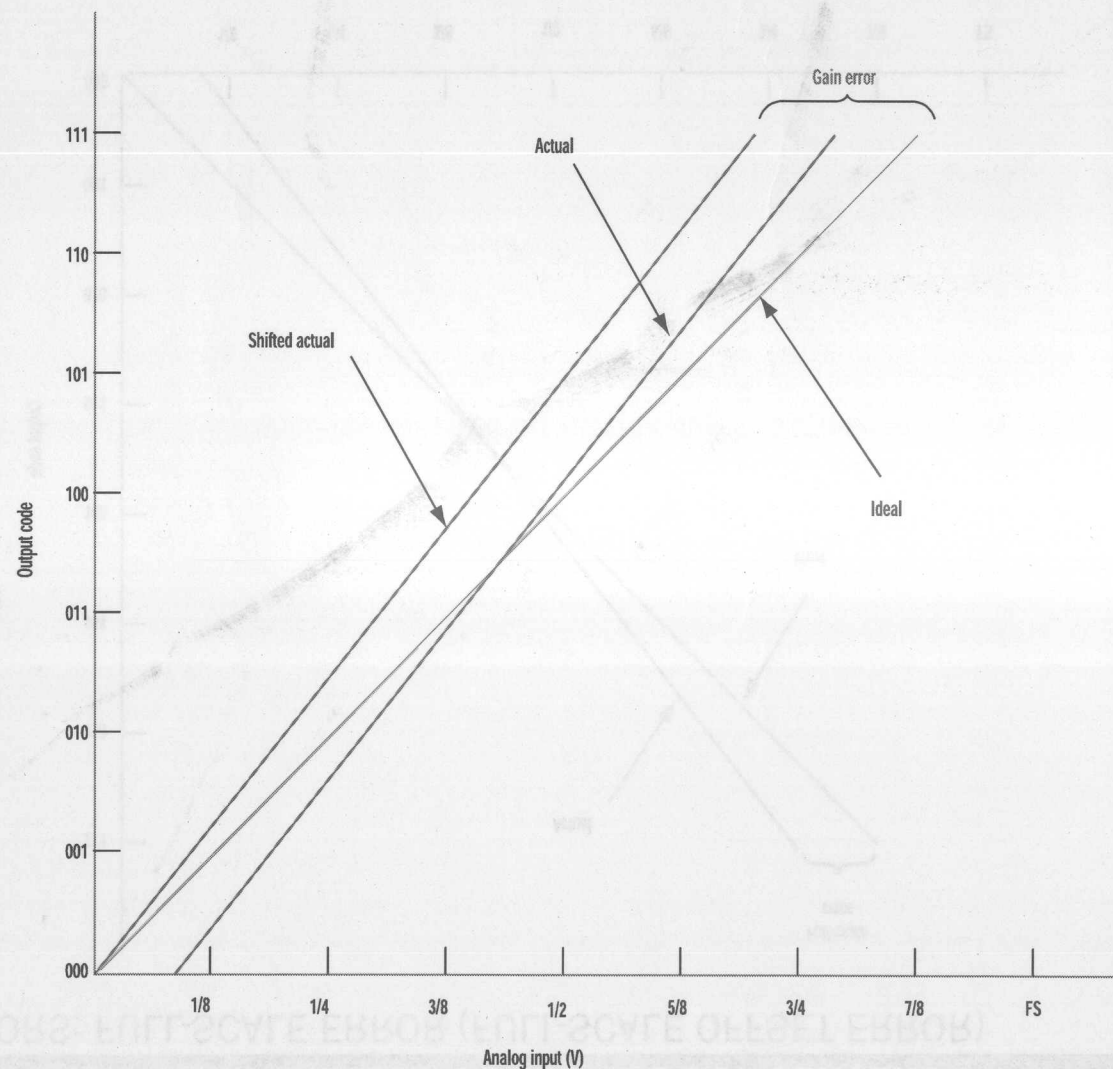
$$\text{Gain} * V_{\text{REF}} * (2^n - 1.5) / 2^n$$

In a practical ADC, the full-scale analog input causing this transition may differ somewhat from this ideal value.



END-POINT ERRORS: GAIN ERROR (FULL-SCALE GAIN ERROR)

Gain error, or full-scale gain error, is a deviation from the ideal slope of the transfer function. It is the same as full-scale error with the offset error subtracted. Shifting the actual transfer curve so the offset error becomes zero, the difference between the actual and ideal transitions to full scale is the gain error. Gain error is expressed in LSBs, in volts, or as a percentage of the ideal full-scale voltage.



electronic design

Sponsored by

 **National**
Semiconductor

 **National**
Semiconductor

A BLACK-BOX APPROACH To Analog-To-Digital Converters

Chapter 3

Don Tuile, *Electronic Design* and Nick Gray, *National Semiconductor*

electronic
design  **eBook**

NEXT PAGE

+ + + A BLACK-BOX APPROACH TO ANALOG-TO-DIGITAL CONVERTERS

TABLE OF CONTENTS: CHAPTER 3: DYNAMIC PERFORMANCE PARAMETERS

- 03 • INTRODUCTION**
- 04 • SIGNAL-TO-NOISE RATIO (SNR)**
- 05 • APERTURE JITTER (INTERNAL NOISE)**
- 06 • CLOCK JITTER (ONE SOURCE OF EXTERNAL NOISE)**
- 07 • CLOCK JITTER (CONTINUED)**
- 08 • SIGNAL JITTER**
- 09 • CAPACITANCE EFFECTS ON ADC OUTPUTS**
- 10 • DISTORTION**
- 11 • TOTAL HARMONIC DISTORTION (THD)**
- 12 • TOTAL HARMONIC DISTORTION (CONTINUED)**
- 13 • SIGNAL-TO-NOISE AND DISTORTION (SINAD)**
- 14 • EFFECTIVE NUMBER OF BITS (ENOB)**
- 15 • SPURIOUS-FREE DYNAMIC RANGE (SFDR)**
- 16 • INTERMODULATION DISTORTION (IMD)**
- 17 • CONCLUSION**

INTRODUCTION

Chapter 3 explains the implications of six important characteristics of analog-to-digital converters (ADCs) for accuracy of measurement:

SNR: signal-to-noise ratio

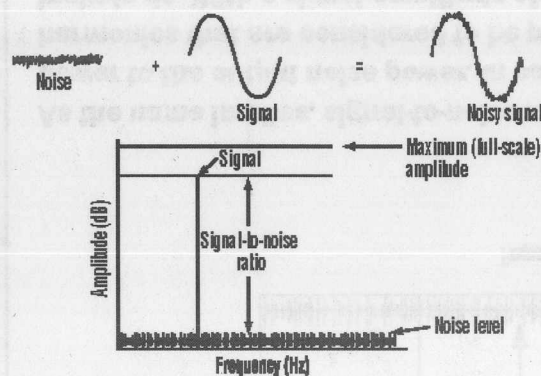
THD: total harmonic distortion

SINAD: signal-to-noise and distortion

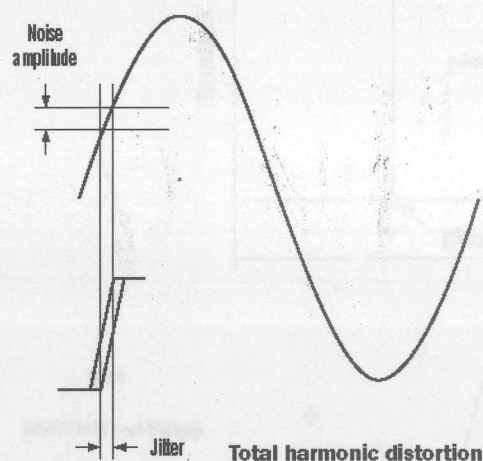
ENOB: effective number of bits

SFDR: spurious-free dynamic range

IMD: intermodulation distortion



Signal-to-noise ratio

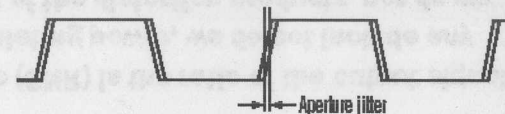


$$\text{SINAD} = -20 \cdot \log \sqrt{(10^{\frac{\text{THD}}{20}})^2 + (10^{\frac{\text{SNR}}{20}})^2}$$

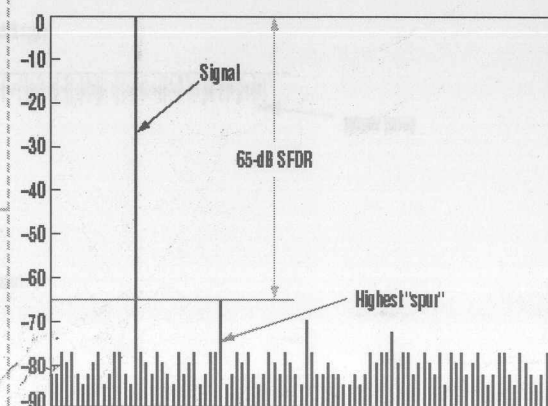
$$\text{SINCE } (10^{\frac{\text{THD}}{20}})^2 = (10^{\frac{\text{THD}}{10}})$$

$$\text{SINAD} = -20 \cdot \log \sqrt{10^{\frac{\text{THD}}{10}} + 10^{\frac{\text{SNR}}{20}}}$$

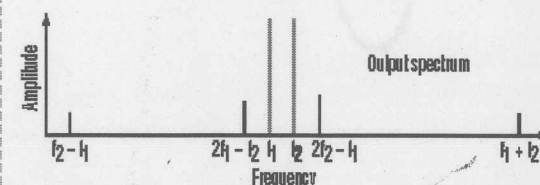
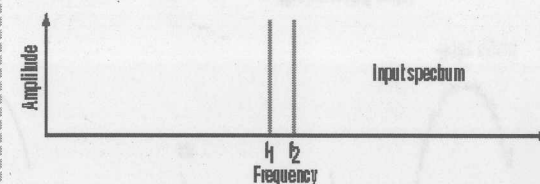
Signal-to-noise and distortion



Aperture jitter



Spurious-free dynamic range



Intermodulation distortion

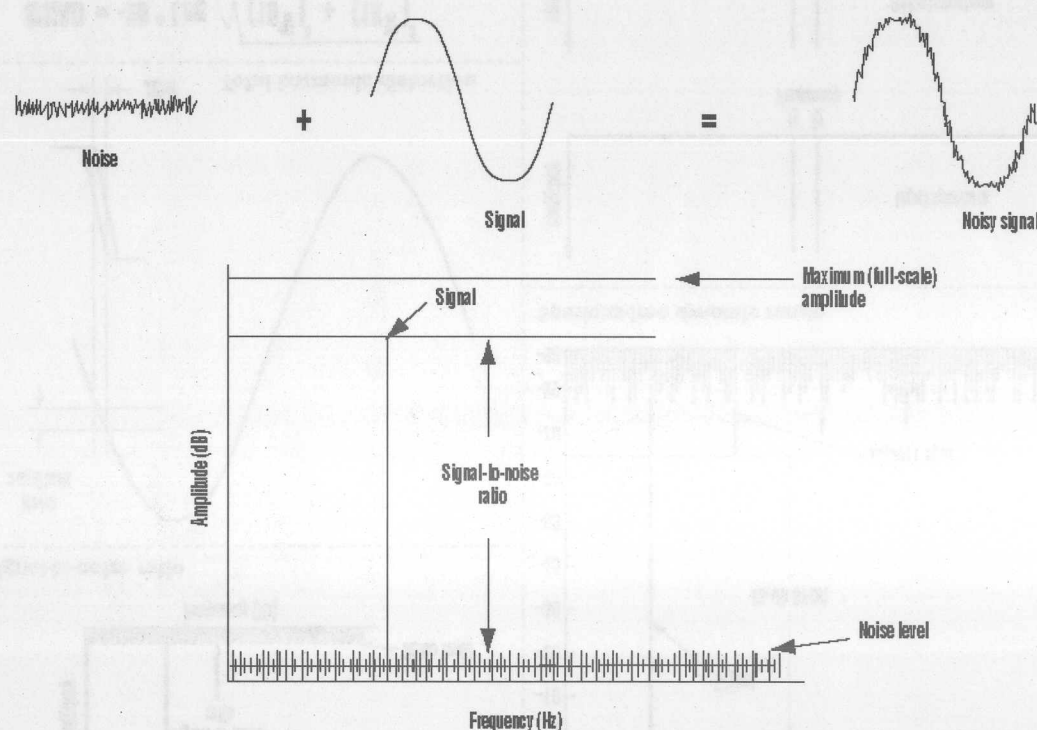
SIGNAL-TO-NOISE RATIO (SNR)

As input frequency increases, beyond a certain point, an ADC's signal-to-noise ratio normally degrades. This is a consequence of the higher input slew rates, working with the ADC aperture jitter, and other sources of jitter (or phase noise), which produce higher levels of sample timing uncertainty. This timing uncertainty is rendered as noise at the ADC output.

In an ADC, noise comes from three sources: noise generated by the converter itself, quantization noise, and application-circuit noise.

Quantization noise (discussed in chapter 2) is a way of expressing the effects of quantization error upon the conversion process. The amplitude of the quantization noise decreases as resolution increases because the size of an LSB, or quanta, gets smaller.

Quantization noise limits the maximum SNR for the ADC. The SNR limit for an ideal ADC with a full-scale sine-wave input comes from quantization noise, and it is defined as $20\log(2^{(n-1)} * \sqrt{6})$, or about $6.02n + 1.76$ dB, where "n" is the ADC resolution in bits.



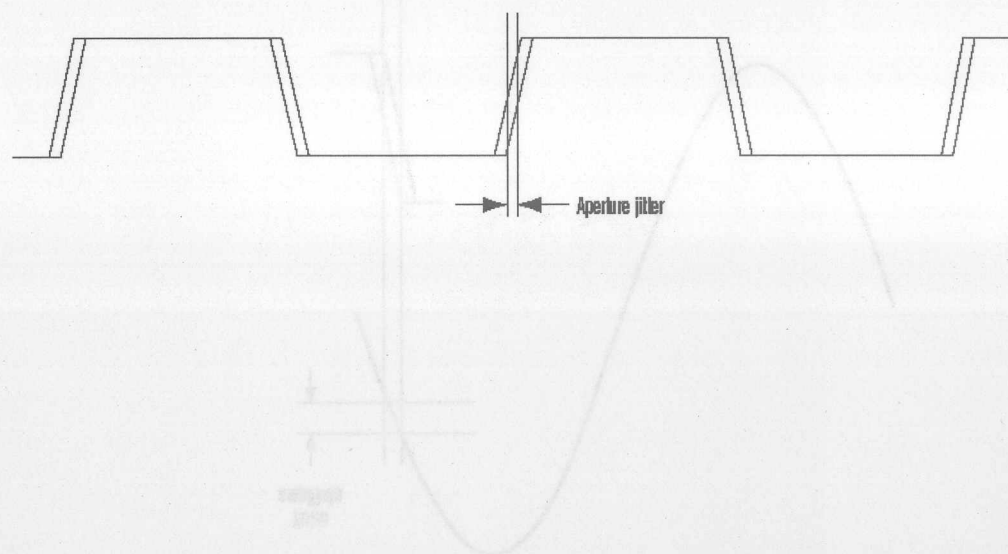
As the name implies, signal-to-noise ratio (SNR) is the ratio of the output signal power to the output noise power. In calculating power, we do not include any harmonics that are considered to be part of the distortion products, nor do we include dc. With a signal amplitude of $1 V_{RMS}$ and a noise level of $100 \mu V_{RMS}$, the SNR is $1 V / 100 \mu V = 10^4$ or 80 dB. Noise level is integrated over the frequency band of interest—usually one-half the clock frequency.

APERTURE JITTER (INTERNAL NOISE)

Aperture jitter usually dominates noise generated by the converter, although the converter's thermal noise also may be a factor.

Aperture jitter is the time variation—on the ADC chip—of the threshold-crossing of the clock signal with a perfectly stable, jitter-free external clock source. It's usually specified in terms of picoseconds rms.

Excessive aperture jitter will degrade the ADC's SNR performance. The design and layout of the ADC die and the die fabrication process both affect aperture jitter.



Aperture jitter is the time variation of the internal threshold-crossing of the clock relative to the signal.

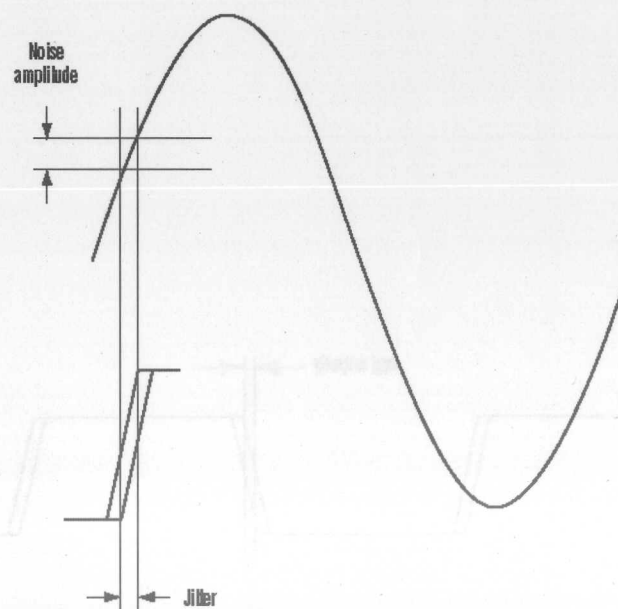
CLOCK JITTER (ONE SOURCE OF EXTERNAL NOISE)

Application circuit noise is noise from the ADC that results from the way the circuit is designed and laid out.

Sources include clock-to-signal jitter, excessive digital output load capacitance, inadequate supply and/or reference bypassing, poor grounding, inattention to signal-integrity issues, poor pc-board layout, and poor design of support circuitry.

This chapter will consider clock and signal jitter, along with circuit-capacitance effects. Other sources of application-circuit noise are beyond the scope of this eBook.

The illustration shows that as clock timing changes slightly, there is a change in the exact point of sampling. This causes the ADC to sample a higher or lower point of the signal than it should. The next point could have a different time variation and a different change from the proper sampling point. The result is variation in the signal sampling point and degraded ADC SNR performance.



Max jitter = $(V_{IN}/V_{FS}) * 1/(2^{(n+1)} \pi f_{IN})$, where:

V_{IN}/V_{FS} is the input amplitude relative to full scale

n is ADC resolution in bits

f_{IN} is the ADC input frequency

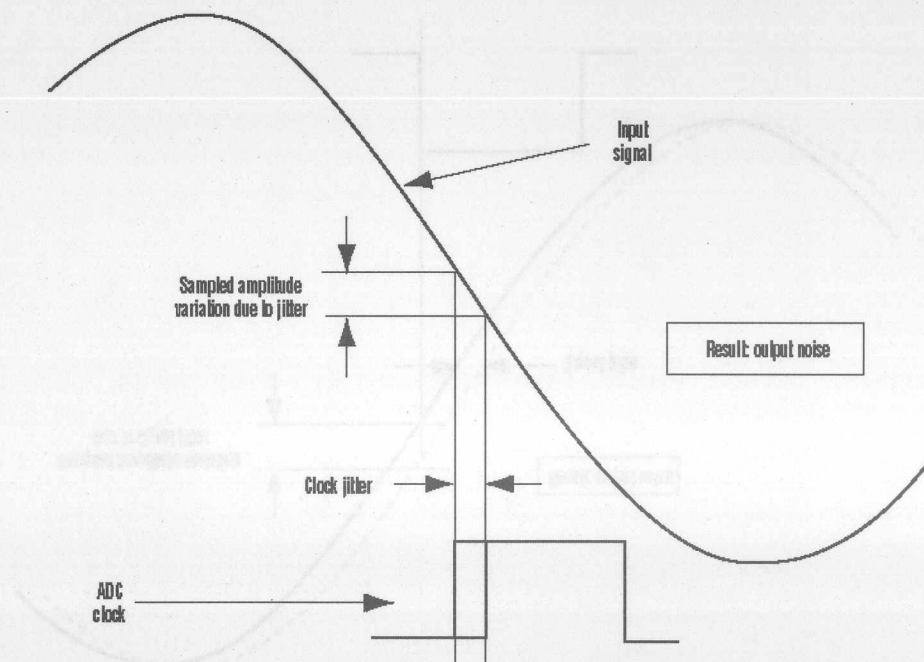
Sometimes this formula is written with 2^n rather than $2^{(n+1)}$. Using n allows for a maximum error of 1 LSB, whereas using $n + 1$ allows for a maximum error of 1/2 LSB. The former leaves some small degradation of SNR, whereas the latter statistically will leave no error due to clock jitter. To prevent noise degradation, the allowable jitter is determined only by the input amplitude relative to full scale, the ADC resolution in bits, and the input frequency. Sample rate has no effect upon ADC output noise.

CLOCK JITTER, CONTINUED

As noted on the previous page, jitter in the ADC clock relative to the input signal results in a variation in the time a signal is sampled, causing a variation from ideal in the sampled signal level.

The higher the ADC resolution (digital output word width) and the lower the reference voltage (or the higher the signal voltage), the greater the effect on noise.

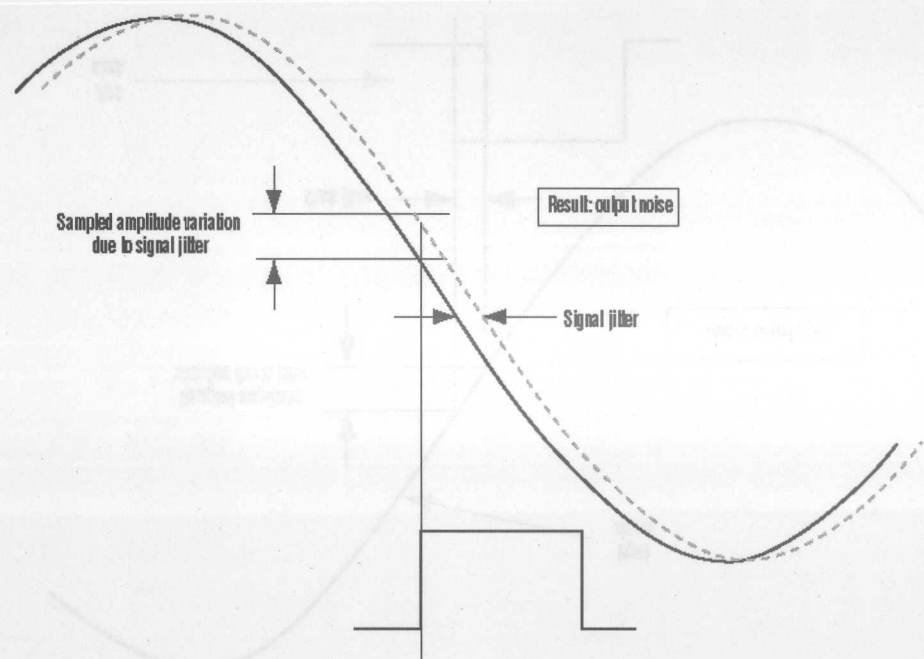
The effect of jitter upon SNR also is more pronounced at higher input signal frequencies. The illustration on the right offers a numerical example.



If jitter is present and the ADC tries to sample the same point in a repetitive waveform at every cycle of that waveform, it may sample levels around the desired voltage. For example, if jitter uncertainty causes a 1.14- to 1.20-V or a 60-mV spread, there is 60 mV of noise at the output of the ADC. Consider an 8-bit ADC and a 12-bit ADC, each with a 2.0-V reference. 1 LSB on the 8-bit ADC represents 7.8 mV, and 1 LSB on the 12-bit ADC represents 0.49 mV. Obviously, the effect of jitter is much more obvious in the higher-resolution ADC.

SIGNAL JITTER

When considering jitter in ADC applications, we generally think of clock jitter. However, jitter added to the input signal by poorly designed signal-path circuitry has the same effect as jitter in the sample clock, as indicated here. Clock jitter relative to the signal jitter really defines the problem. Both clock jitter and signal play a part.



Signal jitter has the same effect as clock jitter.

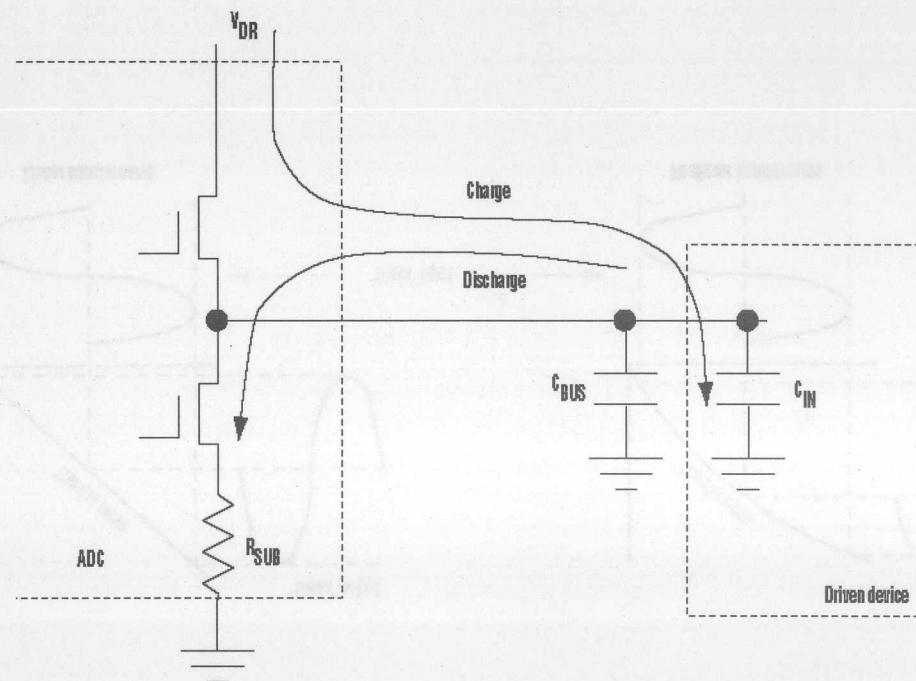
CAPACITANCE EFFECTS ON ADC OUTPUTS

If the impedance of the supply line to an ADC is too high, the voltage on the supply pin may bounce around at the output data rate.

If the ADC doesn't have sufficient ac power-supply rejection ratio (PSRR, a standard datasheet characteristic) at the frequency corresponding to the output data rate, that supply variation can get into the analog portion of the ADC, resulting in noisy conversions.

Ground bounce (described on the right) is another noise source that results from poor circuit design. The most efficient way to reduce ground-bounce currents is to minimize the external load capacitance.

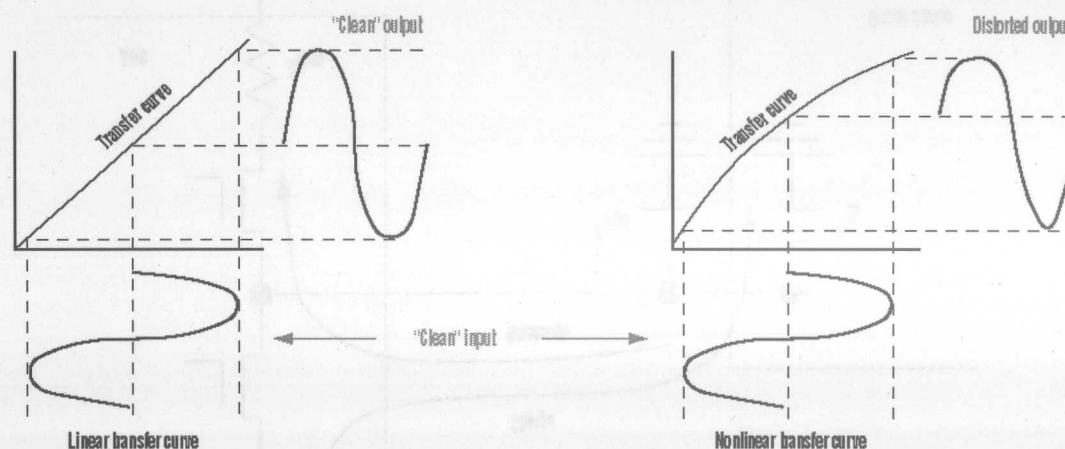
Supply bypassing also is necessary to stabilize the supply voltage.



Charging bus and driven device input capacitances cause noise on the supply line because the outputs require current to charge these capacitances. Discharging these capacitances into the die when one or more outputs go from a high to a low can inject noise into the ADC substrate, causing ground bounce on the die, even though the external ground pins are very quiet. Ground bounce can appear at the input as noise.

DISTORTION

If the input/output transfer curve of any device (ADC or otherwise) is nonlinear, the output of that device will be distorted, as shown by the exaggerated nonlinear transfer characteristic on the right. Chapter 2 of this eBook discusses integral nonlinearity (INL) and the fact that it is related to THD. The illustration shows just how that nonlinearity produces distortion.

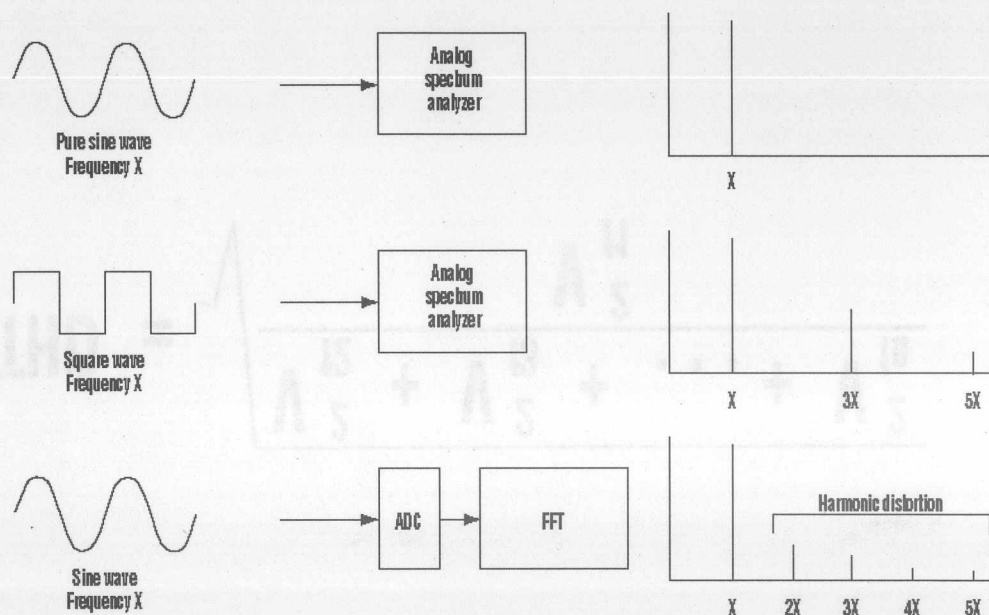


TOTAL HARMONIC DISTORTION (THD)

THD indicates a circuit's linearity in terms of its effect on the harmonic content of a signal. In the ideal case, a signal processing system won't add or subtract any harmonic components—unless that's the signal processor's intended function.

THD performance degrades as the input frequency increases beyond the point at which the input circuitry of the ADC becomes slew-limited.

As a practical matter, there is no such thing as a completely linear input to output transfer function, and any nonlinearity leads to the production of some output distortion. Yet distortion can be so low, it becomes negligible.



An ideal, spectrally pure sine wave (top) has one frequency component. That's all that would appear if we applied that pure sine wave to the input of a spectrum analyzer. A square wave display (middle) would contain odd harmonics with various amplitude and phase relationships. Nonlinearity in an ADC's transfer function will produce harmonics that weren't present in the original signal. We could extract these harmonics if we performed a Fourier transform (FT) upon the ADC output (bottom).

TOTAL HARMONIC DISTORTION, CONTINUED

THD is the ratio of the RMS total power in a specified number of harmonics to the RMS power of the fundamental. In the expression on the right, V_{f1} is the fundamental amplitude, V_{f2} is the second harmonic amplitude, and so on. THD can be expressed as a percentage or in dB.

$$\text{THD} = \sqrt{\frac{V_{f2}^2 + V_{f3}^2 + \dots + V_{fn}^2}{V_{f1}^2}}$$

SIGNAL-TO-NOISE AND DISTORTION (SINAD)

SNR is only the ratio of signal to noise. It does not include the harmonics accounted for in THD calculations or any dc component of the input signal.

In contrast, SINAD describes the ratio of the signal to everything else in the spectrum, except dc. It is found by taking the root sum square of SNR and THD voltage levels and converting that to dB.

Since SINAD incorporates both SNR and THD with equal weighting, SINAD is theoretically at its maximum when SNR is equal to the absolute value of THD.

But at all signal levels, the absolute value of THD is generally greater than SNR. A plot of THD and SNR against any variable will show that this equality almost never occurs.

$$\text{SINAD} = -20 * \text{Log} \sqrt{\left(10^{\frac{\text{SNR}}{20}}\right)^2 + \left(10^{\frac{\text{THD}}{20}}\right)^2}$$

$$\text{SINCE } \left(10^{\frac{\text{SNR}}{20}}\right)^2 = \left(10^{\frac{\text{SNR}}{10}}\right)$$

$$\text{SINAD} = -20 * \text{Log} \sqrt{10^{\frac{\text{SNR}}{10}} + 10^{\frac{\text{THD}}{10}}}$$

Signal-to-noise and distortion (SINAD) is defined as the RMS value of the output signal to the RMS value of all the other spectral components below half the clock frequency, including harmonics but excluding any dc signal component. Because it compares all undesired frequency components with the input frequency, SINAD is an overall measure of ADC dynamic performance. SINAD also is called signal-to-noise and distortion ratio (SNDR) and signal-to-noise plus distortion (S/N+D).

EFFECTIVE NUMBER OF BITS (ENOB)

ENOB, also called effective bits, is a specification that helps to quantify dynamic performance. It's an easily grasped way to specify SINAD.

Any particular ENOB specification says that the referenced converter performs as if it were a theoretically perfect converter with a resolution equal to that value. A 7.5-bit ENOB says that, insofar as SINAD is concerned, that converter performs as if it were an ideal 7.5-bit ADC.

ENOB degrades as frequency increases and as input level decreases for the same reasons that THD and SNR degrade with frequency increases and improve as input level increases.

$$\text{ENOB} = \frac{\text{SINAD} - 1.76}{6.02}$$

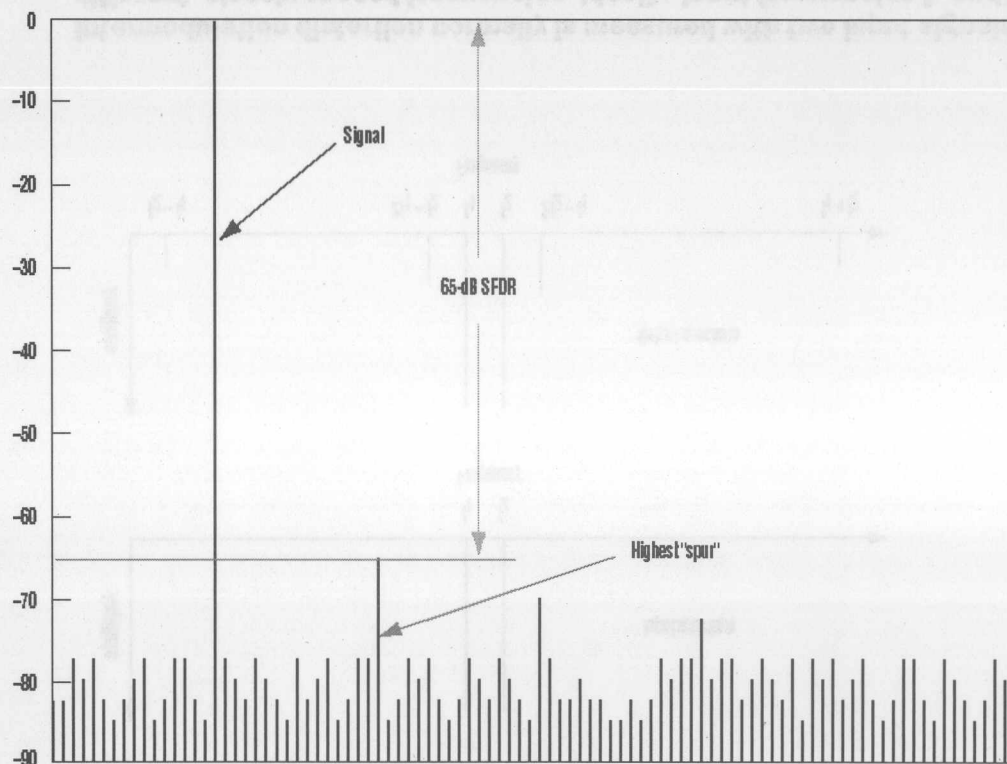
An ideal (perfect) ADC has absolutely no distortion, and the only noise it exhibits is quantization noise. So, its SNR and SINAD are the same. Knowing that SNR for an ideal ADC is $(6.02n + 1.76)$ dB, we can substitute ENOB for n and calculate ENOB according to the formula above, where SINAD is expressed in dB.

SPURIOUS-FREE DYNAMIC RANGE (SFDR)

In the presence of strong signals, nonlinearities in an ADC can produce spurious outputs in the form of frequency components that can mask actual components of very weak signals.

SFDR tells RF designers how weak a signal (in dB relative to a strong signal) their systems can reliably handle. To determine SFDR, designers must look at the entire first Nyquist band. This is because the strongest spur isn't necessarily the one closest to the fundamental.

SFDR is expressed in dB below the fundamental. So, some datasheets show it with a minus sign. But since it is a range, it should be expressed as positive dB. Also, SFDR shouldn't be confused with input dynamic range.

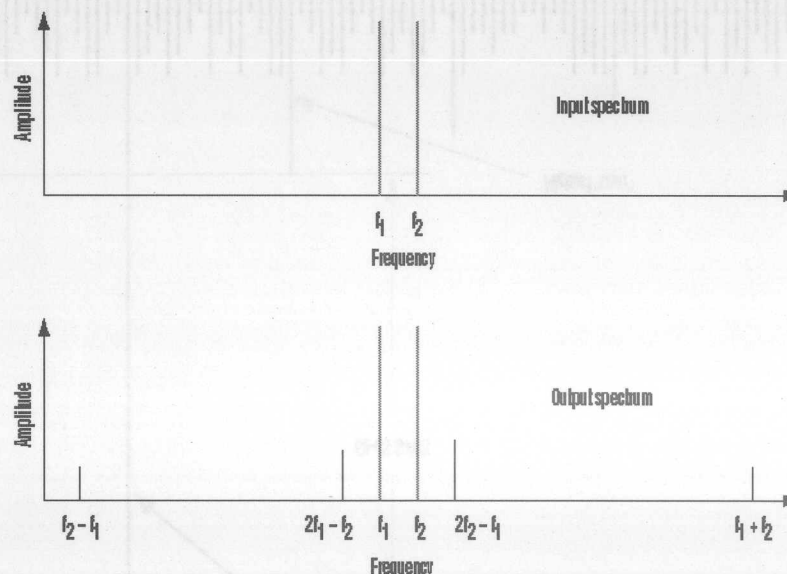


SFDR is the difference between the level of the desired output signal and the level of the highest amplitude output frequency that isn't present in the input, expressed in dB. Some ADC suppliers ignore harmonics when specifying SFDR, but this practice is valid only if those harmonics were present at the ADC input.

INTERMODULATION DISTORTION (IMD)

In contrast to SFDR, which is concerned with spurious responses to a single input signal, IMD reflects an ADC's response to multiple signals. IMD is expressed as the ratio of the power in the intermodulation products to the power in the original input frequencies.

Some applications, particularly those concerned with RF signal processing, are more sensitive to some intermodulation products than to others. In RF applications, the third-order products ($2f_1 - f_2$ and $2f_2 - f_1$) are important because they're closest to the input frequencies, where other terms that are farther from the input frequencies can be digitally filtered out. That's why products other than third-order products often are ignored where IMD is specified for these applications.



Intermodulation distortion normally is measured with two input signals at different, closely spaced frequencies. Ideally, input frequencies f_1 and f_2 should produce output frequencies of only f_1 and f_2 . Device nonlinearities, though, cause the production of new frequencies at the sum and difference frequencies of the input signals and their harmonics, i.e., $(f_1 + f_2)$, $(f_2 - f_1)$, $(f_1 + 2f_2)$, $(2f_1 + f_2)$, $(2f_1 - f_2)$, $(2f_2 - f_1)$, etc. That is, the frequencies (intermodulation products) produced are $\Sigma(mf_1 + nf_2)$, where m and n can take on any integer values. The amplitudes of the intermodulation products will depend upon the nature of the nonlinearity involved.

CONCLUSION

In this chapter, we've defined and discussed the dynamic performance parameters of the ADC and the converter performance implications for these parameters. This eBook series was intended to help readers gain a basic understanding of the ADC without regard to exactly how it performs the conversion function. This knowledge should provide a solid foundation for further, more in-depth study of the ADC.

